CLAIMS

What Is Claimed Is:

1. A method for automatically generating a Function Point Count for a software application, the method being implemented in a programmed computer comprising a processor, at least one data storage system, at least one input device and at least one output device, the method comprising the steps of:

generating, by means of the programmed computer, an object model, said object model representing the functionality of the software application, and further comprising a plurality of objects of functionality having boundaries and inter-relationships between said objects;

quantifying said object model, by means of the programmed computer, by assigning a Function Point Count for said object model;

applying the output of said quantifying to at least one of the output devices.

2. The method of Claim 1, wherein:

said quantifying further comprises data transaction characterization and data set characterization.

3. The method of Claim 2, wherein:

said generating comprises generating an object model that further comprises an application boundary, said application boundary representing the boundaries of functionality of the software application; and

said characterizing comprises further characterizing said data sets as either internal logical files or external interface files, said internal logical files being within said application boundary, and said external interface files being external to said application boundary.

4. The method of Claim 3, wherein:

said characterizing comprises further characterizing each said data transaction as either an external input, an external output, or an external inquiry.

5. A method of providing an automated system for software application quantification, the method being implemented in a programmed computer comprising a processor, at least one data storage system, at least one input device and at least one output device, the method comprising the steps of:

generating, by means of the programmed computer, an object model, said object model comprising an application boundary, said application boundary defining the boundary of said software application;

storing said object model in at least one of the data storage systems;

associating, by means of the programmed computer, said object model with the software application;

first detecting, by means of the programmed computer, data transactions that cross said application boundary;

first characterizing, by means of the programmed computer, said detected boundary-crossing data transactions;

second detecting, by means of the programmed computer, and responsive to said first detecting, data sets influenced by said boundary-crossing data transactions;

second characterizing, by means of the programmed computer, and responsive to said first detecting, said influenced data sets;

quantifying, by means of the programmed computer, the size of said software application responsive to said first and second detecting and characterizing; and

applying the output of said quantifying to at least one of the output devices.

6. The method of Claim 5, wherein said first characterizing step further comprises characterizing said detected data transactions as either an external input, an external output, or an external inquiry.

7. The method of Claim 6, wherein said second characterizing step further comprises characterizing a said data set as an external interface file if said detected influencing data transaction crosses said application boundary.

8. A method for automating the process of quantifying a software application, the method being implemented in a programmed computer comprising a processor, at least one data storage system, at least one input device and at least one output device, the method comprising the steps of:

generating, by means of the programmed computer, an object model, said object model comprising an application boundary, said application boundary defining the boundary of the software application;

storing said object model in at least one of the data storage systems;

associating, by means of the programmed computer, said object model with the software application;

first detecting, by means of the programmed computer, data transactions that cross said application boundary;

first characterizing, by means of the programmed computer, said detected boundary-crossing data transactions;

second detecting, by means of the programmed computer, and responsive to said first detecting, data sets influenced by said boundary-crossing data transactions;

second characterizing, by means of the programmed computer, and responsive to said first detecting, said influenced data sets;

quantifying, by means of the programmed computer, the size of said software application responsive to said first and second detecting and characterizing; and

applying the output of said quantifying to at least one of the output devices.

9. The method of Claim 8, wherein said second characterizing step further comprises characterizing a said data set as an external interface file if said detected influencing data transaction crosses said application boundary.

10. The method of Claim 9, wherein said first characterizing step further comprises characterizing said detected data transactions as either an external input, an external output, or an external inquiry.